

Magnet Project API Documentation

Authentication & User Management APIs

Message Format

All API responses now return bilingual messages in both English and Arabic:

```
{  
    "status": "success",  
    "message": {  
        "en": "User registered successfully",  
        "ar": "تم تسجيل المستخدم بنجاح"  
    },  
    "data": { ... }  
}
```

Verification Rules

Phone Number Verification

- **If phone number is provided:** Only the phone is verified (`isPhoneVerified = true, isEmailVerified = false`)
- **If no phone number is provided:** Only the email is verified (`isEmailVerified = true, isPhoneVerified = false`)

This applies to both customer and business registration.

Auth Routes

1. Register User

- **POST /api/auth/register**
- **Description:** Register a new customer user with automatic verification based on phone number presence.
- **Body:**
 - `firstname` (string, required)
 - `lastname` (string, required)
 - `email` (string, required)
 - `phone` (string, optional)
 - `password` (string, required)
 - `country` (string, required)
 - `language` (string, optional, default: 'en')

- **Verification Logic:**
 - If `phone` is provided: `isPhoneVerified = true, isEmailVerified = false`
 - If `phone` is not provided: `isEmailVerified = true, isPhoneVerified = false`
 - **Response:**
 - `201 Created`: User info + JWT token with verification status
-

2. Register Business

- **POST** `/api/auth/business-register`
 - **Description:** Register a new business user (requires approval) with automatic verification based on phone number presence.
 - **Body:**
 - `firstname, lastname, email, password, crNumber, vatNumber, companyName, companyType, country, city, district, streetName, phone` (all required)
 - **Verification Logic:**
 - If `phone` is provided: `isPhoneVerified = true, isEmailVerified = false`
 - If `phone` is not provided: `isEmailVerified = true, isPhoneVerified = false`
 - **Response:**
 - `201 Created`: Business info (under review) with verification status
-

3. Send Email OTP

- **POST** `/api/auth/send-email-otp`
 - **Description:** Send an OTP to a new email (fails if email already exists).
 - **Body:**
 - `email` (string, required)
 - **Response:**
 - `200 OK`: Success message (bilingual)
-

4. Send Phone OTP

- **POST** `/api/auth/send-phone-otp`
 - **Description:** Send an OTP to a new phone (fails if phone already exists).
 - **Body:**
 - `phone` (string, required)
 - **Response:**
 - `200 OK`: Success message (bilingual)
-

5. Confirm OTP

- **POST** `/api/auth/confirm-otp`
- **Description:** Confirm an OTP for any email or phone identifier (works for both registered and non-registered users).
- **Body:**
 - `identifier` (string, required)

- `otp` (string, required)
 - **Response:**
 - `200 OK`: OTP verified successfully (bilingual message)
-

6. Confirm Login OTP

- **POST** `/api/auth/confirm-login-otp`
 - **Description:** Confirm an OTP for registered users and return user data with JWT token (like login API).
 - **Body:**
 - `identifier` (string, required)
 - `otp` (string, required)
 - **Response:**
 - `200 OK`: User info + JWT token (bilingual message)
 - `404 Not Found`: User not found (bilingual message)
-

7. Login

- **POST** `/api/auth/login`
 - **Description:** Login with email/phone and password.
 - **Body:**
 - `identifier` (email or phone, required)
 - `password` (string, required)
 - **Response:**
 - `200 OK`: User info + JWT token (bilingual message)
-

8. Login with OTP

- **POST** `/api/auth/login-with-otp`
 - **Description:** Request an OTP for login (email or phone).
 - **Body:**
 - `identifier` (email or phone, required)
 - **Response:**
 - `200 OK`: OTP sent (bilingual message)
-

9. Forgot Password

- **POST** `/api/auth/forgot-password`
 - **Description:** Change password for authenticated user.
 - **Headers:** `Authorization: Bearer <token>`
 - **Body:**
 - `newPassword` (string, required)
 - **Response:**
 - `200 OK`: Password updated (bilingual message)
-

User Routes

1. Get Current User Profile

- **GET** /api/user/profile
 - **Description:** Get the profile of the authenticated user (**customer only**).
 - **Headers:** Authorization: Bearer <token>
 - **Role:** customer (requires 'customer' role)
 - **Response:**
 - 200 OK: User info
-

2. Update User Profile

- **PUT** /api/user/profile
 - **Description:** Update the profile of the authenticated user (**customer only**). Email and phone cannot be updated through this endpoint.
 - **Headers:** Authorization: Bearer <token>
 - **Role:** customer (requires 'customer' role)
 - **Body:**
 - firstname, lastname, country, language, imageUrl (all optional)
 - **Response:**
 - 200 OK: Updated user info (bilingual message)
-

3. Get All Business Registration Requests

- **GET** /api/user/business-requests?status=pending&page=1&limit=10
 - **Description:** Get paginated business registration requests (admin/employee only).
 - **Headers:** Authorization: Bearer <token>
 - **Query Params:**
 - status (optional, default: 'pending')
 - page (optional, default: 1)
 - limit (optional, default: 10)
 - **Response:**
 - 200 OK: List of businesses (paginated)
-

4. Approve/Reject Business Registration

- **POST** /api/user/business-approval
- **Description:** Approve or reject a business registration (admin/employee only).
- **Headers:** Authorization: Bearer <token>
- **Body:**
 - businessId (string, required)
 - status (string, required: 'approved' or 'rejected')
 - reason (string, optional, required if status is 'rejected')
- **Response:**
 - 200 OK: Business approval status (bilingual message)

5. Get Business Details by ID

- **GET** /api/user/business/:businessId
 - **Description:** Get details of a business user by ID (admin/employee only).
 - **Headers:** Authorization: Bearer <token>
 - **Params:**
 - businessId (string, required)
 - **Response:**
 - 200 OK: Business info
-

6. Get Business Profile (for Business Users)

- **GET** /api/user/business-profile
 - **Description:** Get the profile of the authenticated business user.
 - **Headers:** Authorization: Bearer <token>
 - **Response:**
 - 200 OK: Business user info
-

7. Favourites APIs (Customer Only)

- **POST** /api/user/favourites/:productId
 - **Description:** Add a product to the customer's favourites.
 - **DELETE** /api/user/favourites/:productId
 - **Description:** Remove a product from the customer's favourites.
 - **GET** /api/user/favourites
 - **Description:** Get all favourite products for the customer.
 - **Headers:** Authorization: Bearer <token>
 - **Response:** All responses include bilingual messages
-

8. Cart APIs (Customer Only)

- **POST** /api/user/cart
 - **Description:** Add a product to the customer's cart or increase its quantity.
 - **Body:** { productId, quantity }
 - **DELETE** /api/user/cart/:productId
 - **Description:** Remove a product from the customer's cart.
 - **PUT** /api/user/cart/:productId
 - **Description:** Update the quantity of a product in the customer's cart.
 - **Body:** { quantity }
 - **GET** /api/user/cart
 - **Description:** Get all products in the customer's cart.
 - **Headers:** Authorization: Bearer <token>
 - **Response:** All responses include bilingual messages
-

Product Routes

1. Get Products

- **GET** /api/products
 - **Description:** Get all approved products (public, any user can call).
 - **Response:**
 - 200 OK: List of products
-

2. Add Product

- **POST** /api/products
 - **Description:** Business user adds a new product (pending approval).
 - **Headers:** Authorization: Bearer <token>
 - **Body:**
 - title, description, image, price, rate, amount, inStock (all required except image and rate)
 - **Response:**
 - 201 Created: Product info (pending approval) with bilingual message
-

3. Update Product

- **PUT** /api/products/:productId
 - **Description:** Business user updates their own product (pending approval).
 - **Headers:** Authorization: Bearer <token>
 - **Body:**
 - title, description, image, price, rate, amount, inStock (all optional)
 - **Response:**
 - 200 OK: Updated product info (pending approval) with bilingual message
-

4. Product Approval

- **POST** /api/products/product-approval
 - **Description:** Admin/employee approves or rejects a product.
 - **Headers:** Authorization: Bearer <token>
 - **Body:**
 - productId (string, required)
 - status (string, required: 'approved' or 'rejected')
 - **Response:**
 - 200 OK: Product approval status with bilingual message
-

5. Delete Product

- **DELETE** /api/products/:productId
- **Description:** Business user deletes their own product, or admin/employee deletes any product (sends email to business user on delete).
- **Headers:** Authorization: Bearer <token>

- **Response:**
 - 200 OK: Product deleted with bilingual message
-

Response Examples

Success Response

```
{  
    "status": "success",  
    "message": {  
        "en": "User registered successfully",  
        "ar": "تم تسجيل المستخدم بنجاح"  
    },  
    "data": {  
        "user": {  
            "id": "60f7b3b3b3b3b3b3b3b3b3b3b3",  
            "firstname": "John",  
            "lastname": "Doe",  
            "email": "john@example.com",  
            "phone": "+966501234567",  
            "imageUrl": "https://example.com/profile.jpg",  
            "isEmailVerified": false,  
            "isPhoneVerified": true  
        },  
        "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9..."  
    }  
}
```

Error Response

```
{  
    "status": "error",  
    "message": {  
        "en": "Email already registered",  
        "ar": "البريد الإلكتروني مسجل بالفعل"  
    }  
}
```

Notes

- All endpoints require authentication unless stated otherwise.
- Role-based access is enforced for some endpoints (Admin, Employee, Business, Customer).
- For endpoints that update or approve/reject, validation middleware is used.
- **All API responses now include bilingual messages in English and Arabic.**
- **Verification status is automatically set during registration based on phone number presence.**

- The GET and PUT `/api/user/profile` endpoints are protected by the `requireCustomer` middleware, allowing only users with the 'customer' role to access them.
-

For more details or example requests, refer to the code or request further examples.